RECEIVED
CENTRAL FAX CENTER
DEC 14 2005

Certificate of Mailing/Facsimile 37 CFR 1.8(a)

I hereby certify that this correspondence is being:

_____ deposited with the United States Postal Service as first class mail in an envelope with sufficient postage addressed to:

COMMISSIONER OF PATENTS
P.O. Box 1450
Alexandria, VA 22313-1450

__X__ transmitted by facsimile to 571 273 8300

on December 14 2005.

By David E McConnell

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of	:	Date: November 23, 2005
E. A. Corl et al	:	Group Art Unit: 2161
Serial Number: 09/761,939	:	Examiner: M.R. Filipczyk
Filed: 16 January 2001	:	INTERNATIONAL BUSINESS MACHINES CORPORATION
Title: Method, System and Computer Program Product to Partition Filter Rules	:	Intellectual Property Law Dept. D-YXSA B-002/2 P.O. Box 12195
	:	Research Triangle Park, NC 27709

Declaration of prior invention to overcome cited patent (37 C.F.R. 1.131)

The Commissioner of Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

This declaration is to establish completion of the invention in this application in the United States, at a date prior to October 23, 2000, the effective date of the Lu et al U.S. Patent 6,778,984 cited by the Examiner. This declaration is presented in response to the first Official Action in which the Lu et al patent has been cited.

The persons making this declaration are the inventors.

The attached Invention Disclosure document is submitted as evidence to establish the date of completion of the invention of this application. The dates appearing on the original document have been redacted. However, the declarant states that the redacted dates are well prior to October 30, 2000.

The declarants further state that conception of the invention was followed by due diligence from the time of conception to a time just prior to the effective date of the reference, up to the actual reduction to practice of the invention and the filing of this application.

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

Inventor: Everett Arthur Cord, Jr.

Signature: Everett Arthur Cord Jr

Date: 2 Dec 2005

Inventor: Gordon Taylor Davis

Signature:

Date: 11/22/2005

Inventor: Victoria Sue Thio

Signature:

Date 11/23/2005

Inventor: Colin Beaton Verrilli

Signature:

Date: 12/2/05

Inventor: Avraham Zehavi

Signature:

Date:

Inventor: Gordon Taylor Davis

Signature:

Date:

Inventor: Victoria Sue Thio

Signature:

Date

Inventor: Colin Beaton Verrilli

Signature:

Date:

Inventor: Avraham Zehavi

Signature: Zehavi Avraham

Date: 12/8/2005

Disclosure RAL8-2000-0094

Created By: Clark D Jeffries **Created On:** 2000 10:32:35 AM
Last Modified By: Karen Orzechowski **Last Modified On:** 2000 12:29:27 PM

*** IBM Confidential ***

Required fields are marked with the asterisk (*) and must be filled in to complete the form.

Summary

Status	Sent for Search/Awaiting Results
Processing Location	RAL
Functional Area	MD-GHB RAINIER & Network Hardware and Software, Modems, Internet, LANs, WANs...700
Attorney/Patent Professional	Josh G Cockburn/Raleigh/IBM
IDT Team	Josh G Cockburn/Raleigh/IBM; Kenneth Barker/Raleigh/IBM; Todd Rasmus/Raleigh/IBM; Norm Strole/Raleigh/IBM; Joel Geyer/Raleigh/IBM; Joe Logan/Raleigh/IBM
Submitted Date	2000 10:53:16 AM
Owning Division	MD
PVT Score	39
Incentive Program	
Lab	
Technology Code	

Inventors with Lotus Notes IDs

Inventors: Art Corl Jr/Raleigh/IBM, Gordon Davis/Raleigh/IBM, Clark D Jeffries/Raleigh/IBM, Victoria Thio/Raleigh/IBM, Colin Verrilli/Raleigh/IBM, Abraham Zehavi/Haifa/IBM@IBMIL

Inventor Name > denotes primary contact	Inventor Serial	Div/Dept	Manager Serial	Manager Name
*Corl Jr, Everett (Art)	D56846	29/MZDA	003020	Hackett Jr, James
Davis, Gordon	123169	29/RWBA	56101	Goldberg, William D
Jeffries, Clark D	53701	29/RWLA	56101	Goldberg, William D
Thio, Victoria	810692	29/RWBA	56101	Goldberg, William D
Verrilli, Colin	87152	29/MZDA	003020	Hackett Jr, James
Zehavi, Avram	973703	IVV103	958606	Zodik, Gabi

Inventors without Lotus Notes IDs

IDT Selection

Main Idea

Title of disclosure (in English)

Automatic partitioning of filter rules

Idea of disclosure

1. Describe your invention, stating the problem solved (if appropriate), and indicating the advantages of using the invention.

Administrators can create filter rules for network security. In general, information in the headers of an IP packet is used to make a key (fixed length binary string). The key is tested by a filter rule and if the rule fits, the action of the rule (such as permit or deny passage of the packet) is applied. The rule applies to various fixed length components of the key such as IP Source Address, IP Destination Address, and so

on. The rule might or might not have a restricted set of values in each component (as opposed to all possible binary values of the given length). If the rule has a restricted set of values in a component, then the key fits that component if and only if the binary values of the key lie in the set of rule values. A key fits a rule if and only if all components of the key lie in the respective component ranges of the rule. Sometimes a key fits two or more rules and in that case the administrator must declare priorities among rules that guarantee inconsistent actions will not be the outcome of the several fits. The testing of a key realitive to a set of filter rules and the application of the stored action or actions associated with rules that the key fits is called enforcement of the set of filter rules.

Two rules intersect if at least one key fits both.

If the range of values in a component of a rule is exactly one value, then that component of the rule is called an exact component. If all the components of a rule are exact, then the rule is called an exact rule.

If the range of values in a component of a rule is all possible binary values of the component length, then that component of the rule is called a wildcard component. Values in a wildcard component are ignored, that is, they are not tested when seeking a rule fit.

It can happen that some rules have ranges in two or more components of a key or, having a range in only one component, might not include all possible values in that one range component. Such rules are called herein "range rules." In general, sets of range rules include some rules that intersect. Also, sets of range rules in general are difficult to administer because it can be not obvious which rules apply to various keys. One method of testing keys relative to such sets of intersecting rules is the Software Managed Tree (SMT) method as disclosed in docket RAL-1999-0141 (disclosure RAL8-1999-0293).

The general management of static and dynamic filter rules is taught in docket RAL9-00-0031 (disclosure RAL8-2000-0027).

It can also happen that some rules have ranges on only one component of a key, for example, there might be a thousand rules in which every component is exact except for the Destination Port number in every rule, which is in every one of the thousand rules a wildcard component. Let us call a set of such rules with given common wildcard component "almost-exact rules." In a set of almost-exact rules, no two of any such rules intersect. One method of testing keys relative to such sets of nonintersecting rules is first hashing fixed components of both the rules, preferably a hash like the geometric hash described in docket RA9-98-056 (diclosure RAL8-1998-0087). Then the key can be processed by the Full Match Tree (FMT) method as disclosed in docket RAL9-1999-0139-US1.

It can be proven that no two nonidentical almost-exact rules with the same wildcard component intersect.

If many almost-exact rules are mixed with other rules, then the combined set can be difficult to enforce at high speed.

2. How does the invention solve the problem or achieve an advantage,(a description of "the invention", including figures inline as appropriate)?

The present invention solves the problem of enforcing a mixture of many almost-exact filter rules together with typically a few range rules.

The present invention includes the concept of administrative use of a partitioning mechanism of rules into

1. A set of n = one or more special components such as Destination Port.

2. For each special component, a maximal set of almost-exact rules labeled AE1, AE2, ..., AEn that are fixed in all components except that have a wildcard component in the special component.
3. All complementary set of rules labeled C consisting of all rules that are not include in the disjoint sets of rules AE1, AE2, ..., AEn defined by 1 and 2.

In this embodiment, no almost-exact rule intersects with any other rule.

Given such a partition, the present invention also includes a total of n separate FMTs corresponding to the sets of almost-exact rules AE1, AE2, ..., AEn.

The present invention also includes one SMT for the rules in the complementary set C.

No rule in a set AEi interesects any other rule in any AEj. No rule in any AEi intersects any rule in C. Therefore the above FMT and SMT tests can be carried out in parallel without consideration of priority of rules.

Following is psuedo-code for generating AE1, ..., AEn and C from a general set of filter rules with n components.

Start

1. Let F be the set of all filter rules to be partitioned. Each filter rule has n components with bit lengths B1, B2, ..., Bn.
2. For each i = 1, 2, ..., n let Si be initially an empty set.
3. Test each rule in F for the property that in component I the range of values in the rule is the interval [0, 2^Bi - 1]. Include in set Si if pass.
4. Test each rule that passes 3 for the property that in all components except I the rule is exact. Delete from set Si if fail.
5. Test each rule in set Si for intersection with any rule not in set Si. Delete from Si if at there is at least one such intersection. [Alternatively, compute the priority number of rules in set Si and delete any from Si that are not of priority number 1.]
5. For each i, determine the number Ni of rules in set Si.
6. If Ni >= a threshold (determined in part by the number of rules in F), then declare Si a "partition set."
7. The complement set C is the set of rules in F and not in any Si.
8. Test rules in each Si with the appropriate hash function and FM Tree.
9. Test rules in C with SMT.

****

In an alternative embodiment, all rules in F have permit or deny as action. Use is made of the priority number mechanism in docket 09/312,148 (disclosure RA8980124). If all almost-exact filter rules have priority number 1 and if all FM searches have priority over the SMT search, then the correct filter rule action will be enforced.

3. If the same advantage or problem has been identified by others (inside/outside IBM), how have those others solved it and does your solution differ and why is it better?
We are not aware of others using rule partitions to simplify filter rule tests.

4. If the invention is implemented in a product or prototype, include technical details, purpose, disclosure details to others and the date of that implementation.

The invention is planned to be implemented in Classifier Rules and Statistics (CRS), a software GUI and XML interface for entering and testing filter rules in using the iBM Power Network Processor (Rainier). CRS is intended to be packaged as a demonstration tool. Both it and Power Network Processor Reference Code Kit are targetted for availability by year end 2000. CRS has not been announced.

***Critical Questions (Questions 1 - 7 must be answered)**
Patent Value Tool (Optional - this may be used by the inventor and attorney to assist with the eval
Evaluation
Search Information
Post Disclosure Text & Drawings

(Form Revised 12/17/97)